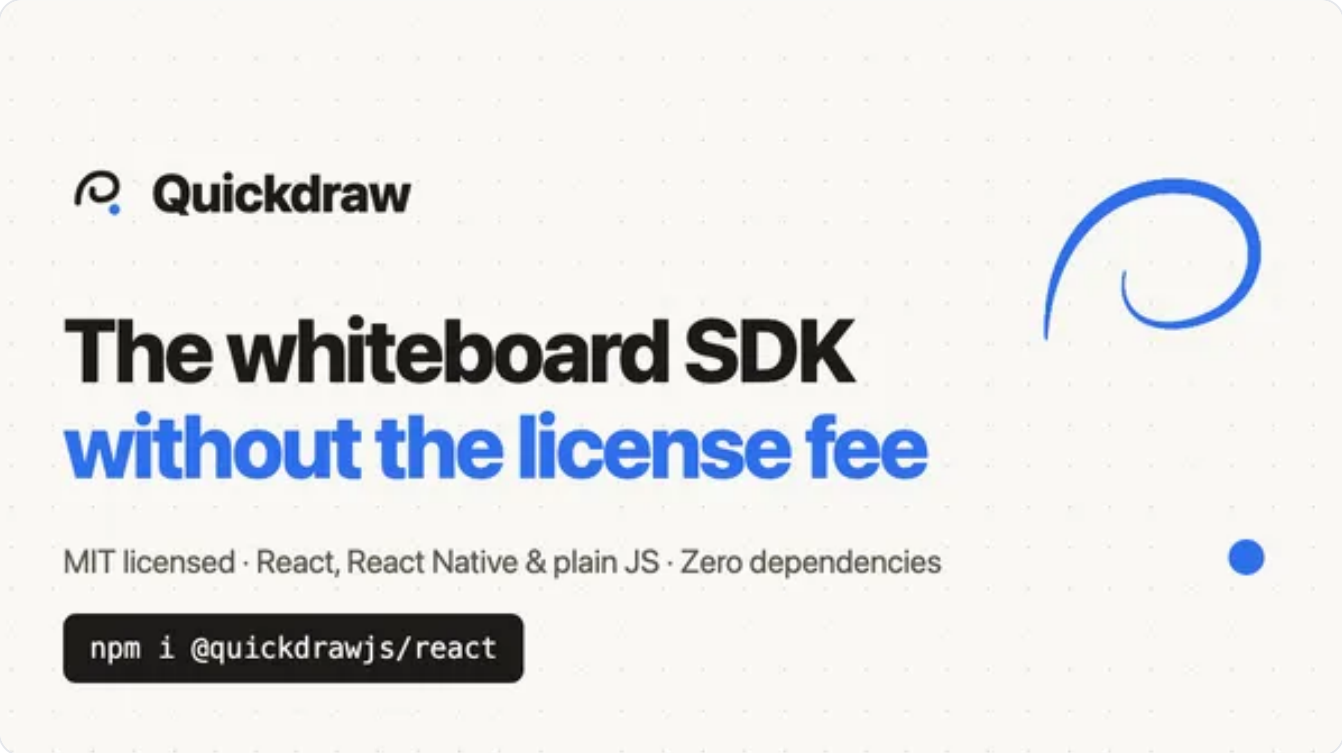
Q

Quickdraw

Free, MIT-licensed whiteboard SDK for React, React Native, and plain JavaScript with zero dependencies.


Quickdraw
The whiteboard SDK
without the license fee
MIT licensed · React, React Native & plain JS · Zero dependencies
npm i @quickdrawjs/react

About

Quickdraw is a whiteboard SDK for React, React Native, and plain JavaScript that ships as a free, MIT-licensed alternative to tldraw. It is built for teams that want an infinite canvas they can embed in products without license fees, license keys, or dependency bloat. The app/site positions Quickdraw as a complete drawing surface with pressure-sensitive ink, hand-drawn shapes, sticky notes, real-time-ready sync, and plain JSON persistence.

What is Quickdraw?

Quickdraw is a whiteboard SDK designed for developers who need an embeddable

Native, and plain JavaScript, so the same core idea can fit a web app, a native app, or a vanilla JS integration.

The project describes itself as a complete infinite-canvas whiteboard for building a whiteboard SDK into product interfaces. It is MIT licensed, free for commercial use, and explicitly states that there is no license fee and no license key. The published bundle size shown on the site is about 25 kB min+gzip, and the core is presented as zero-dependency.

Quickdraw is also framed as an open-source tldraw alternative. That comparison matters because the app is not just a drawing widget; it is meant to be a whiteboard SDK that developers can drop into an application, sync externally, and persist as data.

Key features

Quickdraw focuses on a practical whiteboard SDK feature set rather than a sprawling design suite.

- **Pressure ink**: pen width follows stylus pressure, or velocity for mouse input, so strokes taper in a more natural way.
- **Hand-drawn shapes**: rectangles, ellipses, triangles, diamonds, hexagons, and stars are available with a sketched look, four fill styles, and editable labels.
- **Real-time-ready sync**: every change emits a JSON-safe diff that can be sent over any transport. Remote updates are kept out of the local undo history.
- **Palm rejection**: when a stylus is detected, fingers move the camera while the pen continues drawing, which is especially useful with Apple Pencil-style workflows.
- **Infinite canvas controls**: pan, wheel zoom, two-finger pinch, and zoom-to-fit are all part of the whiteboard SDK experience.
- **Utility tools**: arrows with draggable bends, sticky notes, images, laser pointer, PNG export, per-gesture undo, light and dark themes, and a full keyboard map.
- **Persistence**: store.getSnapshot() returns plain JSON, which makes saving and restoring a board straightforward.

The site also mentions a watermark toggle. Passing `watermark: false` removes the corner mark, and the default watermark appears to help people discover the project.

Using this whiteboard SDK is presented as a simple install-render flow.

1. Install the package for your stack.
   - React: `@quickdrawjs/react`
   - Plain JavaScript: `@quickdrawjs/core`
   - React Native: `@quickdrawjs/react-native`
2. Import the Quickdraw CSS from `@quickdrawjs/core/quickdraw.css`.
3. Mount the component or create the board in a container.
4. For syncing, listen to `onChange` and send the emitted diff wherever your app stores collaboration data.
5. For persistence, call `store.getSnapshot()` and save the returned JSON, then restore later with `store.loadSnapshot(snap)`.

Example usage shown on the site includes a React component, a plain JS `createQuickdraw()` setup, and a React Native example that uses `react-native-webview` and `onChange={(diff) => sync(diff)}`.

Because Quickdraw is a whiteboard SDK, the main setup decision is not how to draw, but where to connect it: local state, database storage, or a collaboration layer you already run.

Who it is for

Quickdraw is aimed at developers and product teams that want a whiteboard SDK they can embed without licensing friction.

- **SaaS teams** that need collaborative boards inside a product and want commercial use covered by MIT.
- **React and React Native developers** who want one drawing surface across web and mobile-adjacent stacks.
- **Frontend engineers** who prefer a framework-free core and plain JSON persistence.
- **Open-source maintainers** who want to inspect, extend, or self-host the code path.
- **Teams comparing alternatives** to tldraw or Excalidraw and prioritizing zero dependencies plus React Native support.

It is especially relevant if you care about a whiteboard SDK that feels lightweight and developer-first rather than a heavy all-in-one canvas suite.

What to know before you use it

Quickdraw is a whiteboard SDK, not a full collaboration backend or document system. The site says sync is real-time-ready, but it does not describe a bundled server, auth layer, or hosted multi-user infrastructure, so you should expect to wire your own transport and persistence.

A few caveats are worth noting:

- The site shows comparisons against tldraw and Excalidraw, but it also says those tools ship more features than Quickdraw does.
- The stated bundle size is a measured figure from the project site, and third-party app builds can differ.
- PNG export, undo behavior, theming, and keyboard mapping are included, but the documentation shown here does not spell out every edge case or supported browser version.
- If you need authoritative collaboration, storage, or audit behavior, you should validate those parts in your own app rather than assuming the SDK handles them.
- If you need license or commercial-use confirmation, the official MIT license text and GitHub repository are the right sources of truth.

In short, Quickdraw is a compact whiteboard SDK for teams that want a permissive, zero-dependency canvas they can shape themselves. If that matches your stack, it is worth trying before you commit to a heavier whiteboard SDK.

FAQ

Is Quickdraw really free for commercial use?

Is Quickdraw free for commercial use?

Can I remove the Quickdraw watermark?

Can I remove the Quickdraw logo?

What platforms does Quickdraw support?

Can I remove the Quickdraw watermark from the canvas?

How do I persist a board?

How do I persist a whiteboard?

How do I save and load a board?

How do I save and restore a board?

How do I save and restore a whiteboard?

How do I save a Quickdraw board?

Does Quickdraw support real-time collaboration?

Does it support real-time collaboration?

Can Quickdraw work with real-time collaboration?

How large is Quickdraw's bundle?

Can I build a custom toolbar?

What is the bundle size of Quickdraw?

How does Quickdraw compare to tldraw and Excalidraw?

Who is Quickdraw best for?

Can I remove the watermark?

Sponsored

Belay – Awa... Featured
Developer Tools

Stays awake while your agents work.

Free # Developer Tools

View ↗

Partly Featured
Finance

Partly is a mobile app that lets friends quickly scan receipts, split any bill fairl...

Free # Finance

View ↗

Visit website ↗

quickdraw

Pricing	Free
Category	Design
Platform	Web, API, React, React Native
Launched	Aug 2026

Verified 1 months ago

TAGS

react # whiteboard sdk # drawing # sdk # infinite-canvas # mit-license
react-native # whiteboard # canvas # javascript # collaboration
open source # real-time # real-time-sync # zero-dependencies

© 2026 Dir Hub

Q Quickdraw